Adaptive Grid Refinement for High-Resolution Finite Element Schemes based on Algebraic Flux Correction

M. Möller & D. Kuzmin
University of Dortmund, Germany

- Algebraic flux correction of TVD type
- Adaptive grid refinement for AFC schemes
  - indicators based on the underlying flux limiter
  - slope-limited gradient recovery techniques
- Application to the compressible Euler equations
- Numerical examples and conclusion
Algebraic Flux Correction of TVD Type  \((Kuzmin, 2004)\)

Scalar conservation law \(\frac{\partial u}{\partial t} + \nabla \cdot (vu) = 0\) in \(\Omega\)

1. Linear high-order scheme (e.g. Galerkin FEM)
   \[ M_C \frac{du}{dt} = Ku, \quad \exists j \neq i : k_{ij} < 0 \]

2. Linear low-order scheme \(L = K + D\)
   \[ M_L \frac{du}{dt} = Lu, \quad l_{ij} \geq 0, \quad \forall j \neq i \]

3. Nonlinear high-resolution scheme \(K^*(u) = L + F^*(u)\)
   \[ M_L \frac{du}{dt} = K^*u, \quad \exists j \neq i : k^*_{ij} < 0 \]

Equivalent LED representation \(L^*u = K^*u\)

\[ M_L \frac{du}{dt} = L^*u, \quad l^*_{ij} \geq 0, \quad \forall j \neq i \]
Linear Low-Order Scheme

Design the artificial diffusion operator $D$ so as to eliminate all negative off-diagonal coefficients

$$M_L \frac{du}{dt} = Ku$$

$$L := K + D$$

$$M_L \frac{du}{dt} = Lu$$

Artificial diffusion

$$(Du)_i = - \sum_{j \neq i} f_{ij}$$

$$f_{ij} = d_{ij}(u_i - u_j)$$

$$f_{ji} = -f_{ij}$$

Optimal diffusion coefficient

$$d_{ij} := \max\{0, -k_{ij}, -k_{ji}\}$$

$$\Rightarrow l_{ij} = k_{ij} + d_{ij} \geq 0$$

Orientation convention: let the edge $\vec{ij}$ be directed so that $l_{ji} \geq l_{ij} = \max\{0, k_{ij}\}$

Local extremum diminishing scheme

$$m_i \frac{du_i}{dt} = \sum_{j \neq i} l_{ij}(u_j - u_i), \quad l_{ij} \geq 0, \forall j \neq i$$

(Jameson, 1993)
Generalized FEM-TVD Formulation

Apply as much antidiffusion $F(u)$ as possible without violating the LED constraint

$$M_L \frac{du}{dt} = Lu \quad K^*(u) := L + F^*(u) \quad M_L \frac{du}{dt} = K^*u$$

Sums of positive/negative diffusive fluxes

$$Q^\pm_i = \sum_{j \neq i} \max\{0, k_{ij}\} \max \{0, u_j - u_i\}$$

Sums of positive/negative antidiffusive fluxes

$$P^\pm_i = \sum_{j \neq i} \min\{0, k_{ij}\} \min \{0, u_j - u_i\}$$

Nodal correction factors

$$R^\pm_i = \Phi(Q^\pm_i / P^\pm_i)$$

computed by a standard TVD limiter $\Phi$

Limited antidiffusion

$$(F^* u)_i = \sum_{j \neq i} f^*_{ij}$$

$$f^*_{ij} = \begin{cases} R^+_i f_{ij} & \text{if } f_{ij} > 0 \\ R^-_i f_{ij} & \text{if } f_{ij} < 0 \end{cases}$$

$$f^*_{ji} = -f^*_{ij}$$
Error Indicators based on Flux Limiters

“The flux limiter implicitly detects structures in the flow that it ‘knows’ cannot be resolved on the underlying grid.”

Net (anti-) diffusion: \[(\Delta F u)_i = \sum_{j \neq i} \Delta f_{ij} \quad \Delta f_{ij} = (1 - R_i)f_{ij} \]

Sums of positive/negative fluxes: \[\Delta f_i^\pm = \sum_{j \neq i} \max \left\{ 0, \Delta f_{ij} \right\} \]

Normalized nodal indicator \[0 \leq \eta_i \leq 1 \]

Choices for patch \(J_i\)
- \(J_i = \{1, \ldots, NVT\}\)
- \(J_i = \{i\} \cup \{j : \exists \vec{i}j \lor \vec{j}i\}\)
- \(...\)

The amount of rejected antidiffusion provides an estimate of the unresolved flow
Example: Solid Body Rotation

$t = 0$

Exact solution / initial data

FEM-TVD solution / superbee limiter

$t = 2\pi$
Error Estimators based on Gradient Recovery

Suppose the true solution is sufficiently smooth ...

Approximate solution: $u_h$ is continuous if $P_1$ trial functions are employed

Piecewise-constant gradient: $\nabla u_h$ is discontinuous across element boundaries

High-order averaged gradient: $G(u_h)$ is a smoothed $C_0$ approximation to $\nabla u$

Recovery-based error estimator: $\eta = \sqrt{\int_{\Omega} |G(u_h) - \nabla u_h|^2 \, dx}$ \hspace{1cm} (Zienkiewicz-Zhu, 1987)

Computation of $G(u_h)$: local $L_2$ projection / superconvergent patch recovery \hspace{1cm} (ZZ, 1990)

What happens if the solution $u_h$ is discontinuous and/or the gradient $\nabla u_h$ changes its sign at local extrema?
Useful Ideas: Finite Volumes

Approximate solution $u_h$ is constant on each cell and discontinuous across element boundaries

Piecewise linear reconstruction

$$\bar{u}(x) = u_i + \sigma_i (x - x_i)$$
on the cell $[x_{i-1/2}, x_{i+1/2}]$

Minmod slope

$$\sigma_i = \text{minmod} \left( \frac{u_i - u_{i-1}}{\Delta x}, \frac{u_{i+1} - u_i}{\Delta x} \right)$$

$$\text{minmod}(a,b) = \begin{cases} a & \text{if } |a| < |b| \land ab > 0 \\ b & \text{if } |b| < |a| \land ab > 0 \\ 0 & \text{if } ab \leq 0 \end{cases}$$

Apply limited averaging to the discontinuous gradient $\nabla u_h$
Error Indicator based on Averaged Gradient Jumps

Define $\eta_{ij}$ as the difference between the ZZ and SL gradients for each edge

1. Recover the **nodal gradient** by a ‘lumped’ $L_2$ projection

   $$(\nabla_h u)_i = \frac{1}{m_i} \sum_{j \neq i} c_{ij} (u_j - u_i)$$

   $m_i = \int_\Omega \varphi_i \, dx$,  $c_{ij} = \int_\Omega \varphi_i \nabla \varphi_j \, dx$

2. Limit the averaged **edge gradient**

   $$(\nabla^* u_h)_{ij} = \minmod(\nabla u^L_h, \nabla u^R_h)$$

3. Estimate the **error** for each edge $i\overline{j}$

   $$\eta_{ij} = \sqrt{\int_{\Gamma_{ij}} |\nabla_h u - \nabla^* u_h|^2 \, dx}$$
Example: Solid Body Rotation

$t = 0$

FEM-TVD solution / superbee limiter

Exact solution / initial data

$t = 2\pi$
‘Pros and Cons’

Error indicator based on the flux limiter

+ available at virtually no additional cost
+ tailored to the treatment of convection dominated flows
+ related to implicit subgrid scale modeling in the MILES context
  – sensitive to the normalization and the choice of the limiter function
  – non-symmetric, i.e., \( K^* \neq -(-K)^* \) and not applicable in the case of \( v \to 0 \)

Error indicator based on the ZZ-SL gradient recovery

+ robust and readily computable
+ detects gradient jumps regardless of the underlying PDE (even for \( v \equiv 0 \))
+ backed by theoretical analysis (ZZ error estimator, DG-FEM / FVM)
Example: Scalar Convection in Space-Time

16,384 (uniform) vs. 2,545 triangles

FEM-TVD solution, superbee limiter, after 3 steps of local refinement
Example: Scalar Convection in Space-Time (cont’d)

FEM-TVD solution, *superbee* limiter, after 6 steps of local refinement

20,543 triangles
Compressible Euler Equations

Divergence form
\[ \frac{\partial U}{\partial t} + \sum_{d=1}^{3} \frac{\partial F^d}{\partial x_d} = 0 \]

Quasi-linear form
\[ A^d = \frac{\partial F^d}{\partial U} \]
\[ \frac{\partial U}{\partial t} + \sum_{d=1}^{3} A^d \frac{\partial U}{\partial x_d} = 0 \]

Lumped FEM discretization
\[ m_i \frac{dU_i}{dt} = \sum_{j \neq i} c_{ij} \cdot \hat{A}_{ij} (U_j - U_i) \]

Cumulative Roe matrices
\[ A_{ij} = a_{ij} \cdot \hat{A}_{ij}, \quad a_{ij} = \frac{c_{ij} - c_{ji}}{2}, \quad B_{ij} = b_{ij} \cdot \hat{A}_{ij}, \quad b_{ij} = \frac{c_{ij} + c_{ji}}{2} \]

Linear low-order scheme
\[ m_i \frac{dU_i}{dt} = \sum_{j \neq i} L_{ij} (U_j - U_i) \]

Edge contribution to the operator \( L \)
\[ L_{ii} = A_{ii} - D_{ij} \quad L_{ij} = -A_{ij} + D_{ij} \]
\[ L_{ji} = A_{ij} + D_{ij} \quad L_{jj} = -A_{jj} - D_{ij} \]

**LED principle for systems:** Render all off-diagonal matrix blocks \( L_{ij} \) positive semi-definite
Design of Artificial Viscosities

Design the artificial viscosity \( D_{ij} \) so as to eliminate negative eigenvalues from \( A_{ij} \)

Characteristic decomposition
\[
A_{ij} = |a_{ij}| R_{ij} \Lambda_{ij} R_{ij}^{-1} \quad |a_{ij}| = \sqrt{a_{ij} \cdot a_{ij}}
\]

Characteristic speeds of wave propagation
\[
\Lambda_{ij} = \text{diag}\{\lambda_i | i = 1, \ldots, 5\}, \quad \lambda_1 = \hat{v}_{ij} - \hat{c}_{ij}, \quad \lambda_2 = \lambda_3 = \lambda_4 = \hat{v}_{ij}, \quad \lambda_5 = \hat{v}_{ij} + \hat{c}_{ij}
\]

Generalized Riemann solver
\[
D_{ij} = |A_{ij}| = |a_{ij}| R_{ij} |\Lambda_{ij}| R_{ij}^{-1}
\]

System `upwinding` (dimensional splitting)
\[
D_{ij} = \sum_{d=1}^{3} |A_{ij}^d| \quad A_{ij}^d = a_{ij}^d \hat{A}_{ij}^d
\]

Scalar dissipation (as a preconditioner)
\[
D_{ij} = d_{ij} I \quad d_{ij} = |a_{ij}| \max_i |\lambda_i|
\]
Characteristic TVD Limiter

In a dimensional splitting approach:

1. Transform the difference of conservative nodal values into characteristic fluxes

2. Compute the sums of raw antidiffusive fluxes for each characteristic field

3. Calculate the nodal correction factors for all local characteristic variables

4. Orient the edges individually for each scalar wave and apply the flux limiter

5. Add the limited antidiffusive correction to the characteristic fluxes

6. Transform the net (anti-) diffusion back to the conservative variables
Example: Compression Corner at $M_{\infty} = 2.5, \quad \alpha = 15^\circ$

- Mach number distribution computed by FEM-TVD
- Unstructured coarse mesh of 403 triangles
- Computational mesh of 11,868 triangles after 6 steps of adaptive grid refinement

Backward Euler pseudo time-stepping, characteristic limiter $\Phi(\theta) = \min(1, 2\theta)$
Example: Scramjet Inlet at $M_\infty = 3$, $\alpha = 0^\circ$

Density isolines

Grid after 3 refinement steps, 74,354 triangles

Mach number isolines

Mid-channel Mach number
Conclusions

- Algebraic flux correction provides a natural framework for adaptive grid refinement

- The amount of rejected antidiffusion yields an estimate of unresolved flow features

- Combination of nodal gradient recovery and edge-based limited averaging leads to the ZZ-SL smoothness indicator

- Error indicators based on flux/slope limiting are
  - robust and readily computable
  - easy to integrate into existing CFD codes
  - good in detecting shocks and discontinuities
In a dimensional splitting approach $d = 1, 2, 3$ do:

1. In a loop over edges compute the left and right eigenvectors $R_{ij}^d$ and $[R_{ij}^d]^{-1}$ for the unidirectional Roe matrix $\hat{A}_{ij}^d$ and the real eigenvalues

$$\lambda_1^d = \hat{\nu}_{ij}^d - \hat{c}_{ij}, \quad \lambda_2^d = \lambda_3^d = \lambda_4^d = \hat{\nu}_{ij}^d, \quad \lambda_5^d = \hat{\nu}_{ij}^d + \hat{c}_{ij}$$

2. Update the sums of edge contributions from the transformed fluxes separately for each characteristic field number $k$ as follows

(a) For $A_{ij}^d$ compute $k_{ij}^a = -a_{ij}^d \lambda_k^d$ and $\delta^\pm_a = \max \min \{0, k_{ij}^a \Delta W_{ij}^k\}$ and update

$$\begin{align*}
P_i^\pm & := P_i^\pm + \delta_a^\pm, \quad Q_j^\pm := Q_j^\pm + \delta_a^\pm \\
P_j^\pm & := P_j^\pm + \delta_a^\pm, \quad Q_i^\pm := Q_i^\pm + \delta_a^\pm
\end{align*}$$

if \begin{align*}
k_{ij}^a < 0 \\
k_{ij}^a > 0
\end{align*}

(b) For $B_{ij}^d$ compute $k_{ij}^b = -b_{ij}^d \lambda_k^d$ and $\delta_b^\pm = \max \min \{0, k_{ij}^b \Delta W_{ij}^k\}$ and update

$$\begin{align*}
P_i^\pm & := P_i^\pm + \delta_b^\pm, \quad P_j^\mp := P_j^\mp - \delta_b^\pm \\
Q_i^\pm & := Q_i^\pm + \delta_b^\pm, \quad Q_j^\mp := Q_j^\mp - \delta_b^\pm
\end{align*}$$

if \begin{align*}
k_{ij}^b < 0 \\
k_{ij}^b > 0
\end{align*}$$

$$\Delta W_{ij} = [R_{ij}^d]^{-1}(U_j - U_i)$$
Characteristic TVD Limiter (cont’d)

3. Compute the nodal correction factors \( R_i^\pm = \Phi(Q_i^\pm / P_i^\pm) \) and determine the ‘upwind node’

\[
I = \begin{cases} 
  i & \text{if } k_{ij}^a \leq 0 \\
  j & \text{if } k_{ij}^a > 0 
\end{cases}
\]

4. Limit the transformed fluxes according to the edge orientation

\[
\Delta \hat{W}_{ij}^k = \begin{cases} 
  R_i^+ \Delta W_{ij}^k & \text{if } \delta_a^k \geq 0, \\
  R_i^- \Delta W_{ij}^k & \text{if } \delta_a^k < 0,
\end{cases}
\]

\[
\delta_a^k = k_{ij}^a \Delta W_{ij}^k
\]

5. Construct the limited antidiffusive fluxes in terms of conservative variables

\[
F_{ij}^d = \left| a_{ij}^d \right| R_{ij}^d | \Lambda_{ij}^d | (\Delta W_{ij} - \Delta \hat{W}_{ij})
\]

6. Initialize \( K^* U := K U \) and apply the limited antidiffusion

\[
(K^* U)_i := (K^* U)_i + F_{ij}^d, \quad (K^* U)_j := (K^* U)_j - F_{ij}^d
\]
Implementation of Characteristic Boundary Conditions

Physical vs. numerical boundary conditions

<table>
<thead>
<tr>
<th></th>
<th>subsonic inflow</th>
<th>supersonic inflow</th>
<th>subsonic outflow</th>
<th>supersonic outflow</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1D 2D 3D</td>
<td>1D 2D 3D</td>
<td>1D 2D 3D</td>
<td>1D 2D 3D</td>
</tr>
<tr>
<td>$N_p$</td>
<td>2 3 4</td>
<td>3 4 5</td>
<td>1 1 1</td>
<td>0 0 0</td>
</tr>
<tr>
<td>$N_n$</td>
<td>1 1 1</td>
<td>0 0 0</td>
<td>2 3 4</td>
<td>3 4 5</td>
</tr>
</tbody>
</table>

Algebraic manipulations for $x_i \in \Gamma$

1. Predict $U_i^* = U_i^{(m)} + \text{diag}\{A_{ii}^{-1}\}R_i^{(m)}$

   and nullify $a_{ij}^{kl} := 0 \quad \forall j \neq i, \forall l \neq k$

2. Transform $U_i^*$ into $W_i^*$ and apply PBC for the incoming Riemann invariants

3. Convert $W_i^{**}$ back to the conservative variables and nullify $R_i^{(m)}$

Variable transformations